MULTIMEDIA UNIVERSITY

STUDENT ID NO | | | | | | | | | | |

MULTIMEDIA UNIVERSITY

FINAL EXAMINATION

TRIMESTER 1, 2018/2019

DCS5038 – PROGRAM DESIGN

(For DIT students only)

23 OCTOBER 2018
9.00 a.m – 11.00 a.m
(2 Hours)

INSTRUCTIONS TO STUDENTS

1. This question paper consists of **ELEVEN (11)** pages excluding the cover page. There are **TWO (2)** sections in this booklet.

2. **SECTION A**: There are **THREE (3)** structured questions. Answer **ALL** questions in the answer booklet provided.

3. **SECTION B**: There are **TWO (2)** questions in this section. Answer **ONLY ONE (1)** question in the answer booklet provided.

SECTION A [75 Marks]

Instruction: Answer **ALL** questions from this section and write your answers in the answer booklet provided.

QUESTION 1 [25 Marks]

a) Draw a complete flowchart based on the following program. (5 marks)

```
#include<tsdio.h>
int function(int x)
{
  int value;
  if (x<=0)
    value = 1;
  else
    value = x - 1 + x;
  return value;
}
int main()
{
   int result;
   result = function(5);
   printf("%d\n",result);
   system("pause");
   return 0;
}
```

b) A company pays its salesperson on a commission basis. In one week, a salesperson receives a weekly pay of RM 215 and a commission of 7% based on his gross sales for that week. For example, a salesperson who sells RM 3000 in a week receives a total of RM 425 as his wages. Write a program based on the case above. The program must :

- Declare all the necessary variables.
- Use preprocessor directives to declare constant values for the commission (7%) and the weekly pay (RM 215).
- Ask the user to input a salesperson's gross sales for the week.
- Calculate and display that salesperson's wages. Here is a sample input/output.

(10 marks)

Continued...

Sample input/output :

```
Enter the gross sales for this week:RM 3000

Gross Sales this week: RM 3000.00
Commission: 7%
Weekly pay: RM215.00
Total wages this week: RM 425.00
```

c) Based on the table given, each product code of a product has its retail price. Write a program that asks the user to enter the product code (*char*) and the quantity sold (*int*) in a day. Use a *switch* statement to determine the retail price for the product. Then calculate and display the total retail value (*float*) of the product sold for the day. (10 marks)

Product code	**Retail price ($)**
A	2.95
R	3.11
Z	4.55

Sample input/output :

```
Enter the product code: R

Quantity sold: 379

The retail price of product R : $3.11
The total retail of the product: $1178.69
```

Continued...

QUESTION 2 [25 Marks]

a) Write the output for the following program. (13 marks)

i.

```
#include<stdio.h>
int Func1(int, int);
void Func2(int);

int main()
{
  int x, y = 4, z = 6, result = 0;
  for(x=0;x<2;x++)
     result = z + Func1(x,y);
  printf("Result is : %d\t%d\n", result,x);
  Func2(result);
  return 0;
}

int Func1(int a, int b)
{
  printf("%d\n",a + b);
  return (b-a);
}

void Func2(int a)
{
   int b = 3;
   printf("%d\n%d\n",a, a*b);
}
```

ii.

```
#include<stdio.h>
int confusion(int a, int b);

int main()
{
  int b = 2, a = 4;
  printf("a = %d   b = %d\n", a, b);
  b=confusion(b,a);
  printf("%d %d\n",a, b);
  return 0;
}

int confusion(int a, int b)
{
  int c;
  printf("a = %d   b = %d\n", a, b);
  printf("%d\n", a*b);
  c = a + b;
  return c;
}
```

Continued...

b) Refer to the following array. The array stores the number of tickets issued by customers in a week via a Service Desk system.

```
int num_ticket [7] = {18,7,15,10,4,6,13};
```

Answer the following questions (*Note: you do not have to write a full program*)

i. Write a code segment that calculates and displays, the total tickets and the average number of tickets issued in that week. Make sure the average is in floating number format of one decimal place. (7 marks)

ii. Write a code segment to find the lowest number of ticket issued in that week and print the number. (5 marks)

QUESTION 3 [25 Marks]

a) Write the content of the array *List* after executing the *for* loop statement. (5 marks)

```
int List[5] = {1, 2, 3, 4, 5 };
    for ( x = 3; x > 0; x-- )
            List[ x ] = 3 * List[ x-1 ];
```

b) Refer to the following structure of MyKad information.

- Birthdate: *birthdate (string)*
- State of birth: *statebirth (string)*
- Last four digits: *last4digits (string)*

Write the code segments based on the following questions.

i. Declare a structure of MyKad information called *MyKad* and a structure variable called *myinfo*. (3 marks)

ii. Ask the user to input data for first, second and last part of *MyKad* and store the data in *birthdate*, *statebirth* and *last4digits* respectively. Below is the sample output screen for your reference. (3 marks)

Continued...

Sample input/output :

```
Enter the first part of your MyKad no: 120125

Enter the second part of your MyKad no: 07

Enter the last part of your MyKad no: 6789
```

iii. Store the last digit of *last4digits* in another *char* variable named *last_char*. Next, convert the character to *integer* data type and store it in a variable name *last_no*. Then write an *if-else* statement to check and display the gender. If the number is even number, display *"Female"*. If the number is odd number, display *"Male"*. Below is the sample output for your reference. (4 marks)

Sample input/output :

```
Last digit is 9
Gender: Male
```

c) Write a program to display the content of **flavor.txt** file. If the file does not exist, display "*File does not exist*" and prompt the user again until the file is successfully read. (10 marks)

Content of flavor.txt file
Chocolate 2.00 Vanilla 1.50 Strawberry 2.50

Sample input/output :

```
Enter file name : flavour.txt
File does not exist

Enter file name : flavor.txt
Chocolate   RM2.00
Vanilla     RM1.50
Strawberry  RM2.50
```

Continued...

SECTION B [25 Marks]

Instruction: Choose and answer **ONLY ONE (1)** question from this section and write your answers in the answer booklet provided.

QUESTION 1 [25 Marks]

A salesperson gets a yearly bonus based on a commission from collections of sales and review from peers. Sales will be calculated for 12 months and the commission is rated based on an average sales table (shown in the next page). Other salesperson in the company will give points for peer review. The average from the peer review determines the review rate.

Write a complete C program based on the above situation and the following instructions.

i. Declare a *structure* of Sales Executive record called *SalesExecutive* that has:
   - Sales Executive name : *exec_name (string)*
   - Basic salary : *basic_salary (float)*
   - Average sales : *averageSales (float)*
   - Average peer review : *averagePeerReview (float)*
   - Total yearly bonus : *yearlyBonus (float)*

ii. In function ***main()*** :
   - Create a structure variable named *staff*
   - Prompt user to enter sales executive name (*exec_name*) and his/her basic salary (*basic_salary*).
   - Call function ***calc_averageSales ()*** to key-in and calculate the average sales obtained in 12 months. Return the average and store the average sales in *averageSales.*
   - Call function ***getCommissionRate (...)*** and pass *averageSales* as the parameter to determine the commission rate. Return the rate.
   - Next, prompt user to enter number of sales executive in the company.
   - Call function ***calc_AveragePeerReview (...)*** and pass the number of the sales executives as entered earlier. Return the average and store it in *averagePeerReview.*
   - Call function ***getReviewRate(...)*** and pass *averagePeerReview* as the parameter to determine the review rate. Return the rate.
   - Next, calculate the amount of bonus based on commission rate.
     - Bonus from commission = commission rate X average of sales
   - Then, calculate the amount of bonus from peer review.
     - Bonus from peer review = review rate X basic salary
   - Then, calculate the total bonus of the year by adding the two bonus amounts.
   - Display all the details of the Sales Executive record as shown in the sample output screens (*Figure BQ1a* and *Figure BQ1b*).

Continued...

iii. In function ***calc_averageSales ():***
- Declare a prototype for this function.
- Create an array to store the collection of sales for 12 months.
- Using *for* loop, prompt the user to enter the collection of sales
  - Store the input in the array accordingly.
  - Calculate total of the sales.
- Then, calculate the average of the sales.
- Return the average sales.

iv. In function ***getCommissionRate(...):***
- Declare a prototype for this function.
- Using *if-else* statements, determine the rate based on the average sales. Refer to the following table:

Average Sales (RM)	Commission Rate (%)
Less than 8000	0
8000 - 15000	10
15999 - 20000	15
More than 20000	20

- Return the commission rate.

v. In function ***calc_AveragePeerReview(...):***
- Declare a prototype for this function.
- Create an array to store the peer reviews based on the number of salespersons.
- Using *for* loop, prompt the user to enter rating from each peer.
  - Store the input in the array accordingly.
  - Calculate total rating of the review.
- Then, calculate the average of the peer review rating
- Return the average peer rating

vi. In function ***getReviewRate(...):***
- Declare a prototype for this function.
- Using *if-else* statements, determine the rate based on the average rating from peers. Refer to the following table:

Average rating	Review Rate (%)
Less than 2.50	0
Less than 3.00	2
Less than 4.00	3.5
4.00 – 5.00	7

- Return the review rate.

Continued...

Sample input/output 1:

```
Enter the Sales Executive name: Lily
Enter the basic salary: RM 2300

Enter the sales for 12 months:
Sales month 1 = 8000
Sales month 2 = 9000
Sales month 3 = 10000
Sales month 4 = 15000
Sales month 5 = 850
Sales month 6 = 10000
Sales month 7 = 7500
Sales month 8 = 9800
Sales month 9 = 10500
Sales month 10 = 10500
Sales month 11 = 8000
Sales month 12 = 12000

Enter the number of sales executives in the company: 4
Enter Review Peer 1 = 4.5
Enter Review Peer 2 = 4
Enter Review Peer 3 = 3.78
Enter Review Peer 4 = 3.89

-------------YEAR 2018 BONUS RECORD----------------
Sales Executive name: Lily      Basic salary: RM2300.00

Average sales:RM 9262.50
Bonus amount from sales:RM 926.25

Average rating of peer reviews: 4.04
Bonus amount from peer reviews:RM 161.00
------------------------------
Total bonus:RM 1087.25
------------------------------
```

Figure BQ1a

Continued...

Sample input/output 2:

```
Enter the Sales Executive name: Jane
Enter the basic salary: RM 2300

Enter the sales for 12 months:
Sales month 1 = 5000
Sales month 2 = 7000
Sales month 3 = 3500
Sales month 4 = 4300
Sales month 5 = 5700
Sales month 6 = 6000
Sales month 7 = 8800
Sales month 8 = 10000
Sales month 9 = 5900
Sales month 10 = 3000
Sales month 11 = 8000
Sales month 12 = 6900

Enter the number of sales executives in the company: 4
Enter Review Peer 1 = 3.5
Enter Review Peer 2 = 2.89
Enter Review Peer 3 = 3
Enter Review Peer 4 = 3

-------------YEAR 2018 BONUS RECORD----------------
Sales Executive name: Jane      Basic salary: RM2300.00

Average sales:RM 6175.00
Bonus amount from sales:RM 0.00

Average rating of peer reviews: 3.10
Bonus amount from peer reviews:RM 80.50
------------------------------
Total bonus:RM 80.50
------------------------------
```

Figure BQ1b

Continued...

QUESTION 2 [25 Marks]

Write a program for the user to input student details and marks. The program will then display the grade based on the mark entered and analyse the highest and lowest mark. Use the following requirements and sample output to write the program.

i. In function ***main()*** :
- Create a file named **student.txt** to store the *name, id and mark.* If the file cannot be created display message *"File could not be opened"*.
- The user can enter as many *name, id* and *mark* as needed.
- Upon completed keying in the details, function ***Display()*** is called.

ii. In function ***Display()*** :
- Retrieve file named **student.txt** and display the information stored. Use *if..else if* statement to display the grade based on the mark entered.

Mark	Grade
mark >= 80	A
mark >= 65	B
mark >= 50	C
mark < 50	F

- While reading the file named **student.txt**, create another file name **mark.txt** to store the student mark only.
- Call function ***Analyse()*** to analyse the result.

iii. In function ***Analyse()*** :
- Retrieve file named **mark.txt** and find the highest and lowest mark. Use *if* statement to find the highest and lowest mark.

Continued...

Sample input/output :

```
Welcome to Mark Entry System

Enter name : Meghan
Enter ID   : F11
Enter mark : 70
Add data? <Y-Yes  N-No> : Y

Enter name : Carlos
Enter ID   : M22
Enter mark : 90
Add data? <Y-Yes  N-No> : Y

Enter name : Catlyn
Enter ID   : F33
Enter mark : 40
Add data? <Y-Yes  N-No> : Y

Enter name : Stephen
Enter ID   : M44
Enter mark : 60
Add data? <Y-Yes  N-No> : N

Grade Report

 Meghan     F11     70     B
 Carlos     M22     90     A
 Catlyn     F33     40     F
 Stephen    M44     60     C

Result Analysis

 Highest mark   :  90
 Lowest mark    :  40
```

End of Page.